Output feedback boundary control of heterodirectional semilinear hyperbolic systems ⋆

Timm Strecker [a], Ole Morten Aamo [b], Michael Cantoni [a]

[a] *Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville 3010, Australia*

[b] *Department of Engineering Cybernetics, Norwegian University of Science and Technology (NTNU), Trondheim N-7491, Norway*

Abstract

We solve the problem of stabilizing a general class of 1-d semilinear hyperbolic systems with an arbitrary number of states convecting in each direction and with the actuation and sensing restricted to one boundary. The control design is based on the dynamics on the characteristic lines along which the inputs propagate through the domain and the predictability of states in the interior of the domain up to the time they are affected by the inputs. In the context of broad solutions, the state-feedback controller drives systems with globally Lipschitz nonlinearities from an arbitrary initial condition to the origin in minimum time. Alternatively, it is possible to satisfy a tracking objective at the uncontrolled boundary or, for systems with C^1-coefficients and initial conditions, to design the control inputs to obtain classical C^1-solutions that also reach the origin in finite time. Further, we design an observer that estimates the distributed state from boundary measurements only. The observer combined with the state-feedback controller solves the output-feedback control problem.

Key words: Hyperbolic partial differential equations, distributed-parameter systems, boundary control, stabilization, estimation

1 Introduction

Hyperbolic partial differential equations are widely used to model systems such as open water channels [15,10], traffic flow [3], flow through pipelines [14] and oil wells [11,1], for the purpose of controller-design. In many situations, actuation and sensing are restricted to the boundary of the spatial domain.

Approaches to the stabilization of nonlinear hyperbolic systems by boundary control include designing dissipative boundary conditions by analysing the evolution of Riemann invariants [13,10] or via a Lyapunov approach [7]. However, these methods do not achieve convergence to a target state in finite time, and there are controllable systems that cannot be stabilized by static boundary feedback [4].

⋆ This work was supported by the Australian Research Council (LP160100666). This paper was not presented at any IFAC meeting. Corresponding author Timm Strecker.

Email addresses: timm.strecker@unimelb.edu.au (Timm Strecker), ole.morten.aamo@ntnu.no (Ole Morten Aamo), cantoni@unimelb.edu.au (Michael Cantoni).

Finite-time or exact controllability, i.e. controlling the state to an equilibrium in finite time, is well established for linear systems [21], semilinear systems (i.e., source terms depend nonlinearly on the state but speeds are independent of state), e.g. [29], and quasilinear systems (i.e., speeds and source terms depend nonlinearly on the state) [6,18]. Controllability is global for linear systems but nonlinear systems are in general only locally controllable, although controllability can be global if additional assumptions are made on the system coefficients, see e.g. [29,15]. Similar results exist for observability [19]. A different form of controllability of such systems is the ability to exactly track a reference signal at, depending on the number of control inputs, one or more locations in the domain, called nodal profile control [14,20].

The controllability result from [18] is constructive as it provides a method for computing the control inputs that drive the system from the initial state to the target state within finite time. This time must be chosen larger than some minimum time depending on the transport speeds. However, it is an open-loop control law, i.e. the control inputs for the whole time interval are computed once based on the initial condition. There is no direct way to update the control inputs based on measurements of the state as time proceeds, making performance and stability sensitive to disturbances and model uncertainty.

A constructive method for developing state-feedback and output-feedback boundary controllers for linear hyperbolic systems has been developed in form of backstepping [22], [28], [16], [2], [8], [1]. However, at this stage backstepping control of hyperbolic PDEs is restricted to linear systems, although it has been shown that quasilinear systems can be locally stabilized by a linear backstepping controller designed based on the linearization of the nonlinear system [9,17].

More recently, a method for designing state-feedback and output feedback boundary controllers for semilinear systems was presented in [23]. The method relies on virtually moving the control inputs to the uncontrolled boundary, and the system can be stabilized by setting these virtual inputs to zero. Based on predictions of the state that cannot be affected due to the system delay, the control inputs are constructed by solving an ODE that governs the dynamics on the characteristic line along which the control input propagates *backwards* in time, taking the virtual input as boundary condition for the ODE. It was shown that for linear systems the state-feedback controller is equivalent to a previously published backstepping controller [23, Section 3.4]. The method was also extended to a class of interconneceted systems [24] and to bilateral boundary control [26].

In this paper, we extend the method from [23] further to systems with m actuated states all convecting in the same direction and n non-actuated states convecting in the opposite direction, for arbitrary n and m. In another sense, it can be seen as an extension of the results in [16,2,8] to semilinear systems. Moreover, rather than only focusing on minimum-time control as in [23], we treat the virtual control inputs as degrees of freedom that can be designed to ensure smoothness of the closed-loop trajectories.

One of the challenges of systems with m inputs rather than only one input as in [23] is the interaction between different control inputs. In particular, the effect a control input associated with a slower propagation speed has on the state depends on future values of control inputs associated with faster speeds (see also Figure 1 for how the inputs propagate through the domain).

This paper is organized as follows. The precise problem statement is given in Section 2. In Section 3 we design the state-feedback control law: some preliminary results on the states and dynamics on the characteristic lines are established in Section 3.1 and a target system for the closed-loop dynamics on the characteristic lines is designed in Section 3.2, before the actual inputs are constructed in Section 3.3. The state estimation problem is solved in Section 4, with the observer given in Section 4.2 and the output feedback control problem solved in Section 4.3. Section 5 shows a numerical example. Concluding remarks are given in Section 6. The appendix contains several technical proofs.

2 Problem statement

2.1 System description

We consider systems in the form

$$\begin{aligned}
u_t(x,t) &= -\Lambda^u(x)u_x(x,t) + F^u((u,v)(x,t),x,t), && (1)\\
v_t(x,t) &= \quad \Lambda^v(x)v_x(x,t) + F^v((u,v)(x,t),x,t), && (2)\\
u(0,t) &= f(v(0,t),t), && (3)\\
v(1,t) &= U(t), && (4)\\
u(x,0) &= u_0(x), && (5)\\
v(x,0) &= v_0(x), && (6)
\end{aligned}$$

where $x \in [0,1]$, $t \geq 0$ and

$$u(x,t) = \Big(u_1(x,t) \; \dots \; u_n(x,t)\Big)^T, \tag{7}$$

$$v(x,t) = \Big(v_1(x,t) \; \dots \; v_m(x,t)\Big)^T, \tag{8}$$

$$U(t) = \Big(U_1(t) \; \dots \; U_m(t)\Big)^T. \tag{9}$$

The subscripts $_x$ and $_t$ denote partial derivatives with respect to x and t, respectively. Throughout the paper, we use $h([a,b])$ to denote a function h (usually state or input) evaluated over the closed interval $[a,b]$ and similarly for the open interval (a,b) and half open intervals $[a,b)$ and $(a,b]$, and $(u,v)(x,t)$ to denote the whole state consisting of both u and v evaluated at (x,t). We assume $u_0 \in \mathcal{X}^n_{[0,1]}$ and $v_0 \in \mathcal{X}^m_{[0,1]}$ where $\mathcal{X}^k_{[a,b]}$ for $k \geq 1$ is the space of bounded vector-valued functions on $[a,b]$,

$$\mathcal{X}^k_{[a,b]} = \left\{ f : [a,b] \to \mathbb{R}^k : \|f\|_\infty < \infty \right\} \tag{10}$$

with norm

$$\|h\|_\infty = \sup_{x\in[a,b]} \|h(x)\| \tag{11}$$

for $h \in \mathcal{X}^k_{[a,b]}$ with Euclidean norm $\|\cdot\|$.

Remark 1 *We consider a type of non-classical solutions of the PDE model (1)-(6) that is defined via the associated integral equations. That is, the PDEs can be transformed to integral equations as given in Appendix A, and we say that the solution of the integral equations, which always exists for bounded inputs and is unique under the given assumptions, is the solution of the PDEs. This solution type has been called broad solutions in [5], although in an L^∞ setting where solutions are defined only almost everywhere as opposed to point-wise. As the solution of* *the integral equations might not be differentiable, depending on initial conditions and control input, the original PDEs might not be satisfied in the classical sense.*

The transport speeds

$$\Lambda^u(x) = \operatorname{diag}\left(\lambda_1^u(x) \ \dots \ \lambda_n^u(x)\right), \tag{12}$$

$$\Lambda^v(x) = \operatorname{diag}\left(\lambda_1^v(x) \ \dots \ \lambda_m^v(x)\right), \tag{13}$$

are assumed to be measurable in x and satisfy

$$\lambda_1^u(x) > \lambda_2^u(x) > \ldots > \lambda_n^u(x) > 0, \tag{14}$$

$$\lambda_1^v(x) > \lambda_2^v(x) > \ldots > \lambda_m^v(x) > 0, \tag{15}$$

for all x.

Remark 2 *The case where two or more states have equal speeds $\lambda_i^{u/v}$ can be handled by allowing the corresponding state u_i or v_i, respectively, to be vector-valued. Further, note that Assumption (14) is only required for the observer design while Assumption (15) is only required for the state-feedback controller design.*

The nonlinearities $F^u, F^v : \mathbb{R}^{n+m} \times [0,1] \times \mathbb{R} \to \mathbb{R}^n$ and $f : \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$ are assumed to be (i) globally Lipschitz-continuous in the state,

$$\begin{aligned} &\|F^u((y_1,z_1),x,t) - F^u((y_2,z_2),x,t)\| \\ &\qquad \le L_u(\|y_1 - y_2\| + \|z_1 - z_2\|), \end{aligned} \tag{16}$$

$$\begin{aligned} &\|F^v((y_1,z_1),x,t) - F^v((y_2,z_2),x,t)\| \\ &\qquad \le L_v(\|y_1 - y_2\| + \|z_1 - z_2\|), \end{aligned} \tag{17}$$

$$\|f(z_1,t) - f(z_2,t)\| \le L_f\|z_1 - z_2\|, \tag{18}$$

for all $y_1, y_2 \in \mathbb{R}^n$, $z_1, z_2 \in \mathbb{R}^m$, $x \in [0,1]$ and $t \ge 0$, (ii) measureable and uniformly bounded in x and t, and (iii) such that

$$F^u((0,0),x,t) = 0, \tag{19}$$

$$F^v((0,0),x,t) = 0, \tag{20}$$

$$f(0,t) = 0, \tag{21}$$

for all x and t.

Remark 3 *The global Lipschitz conditions (16)-(18) ensure global existence of the solution as long as the control inputs remain bounded. Similar local results can be obtained if the nonlinearities are only locally Lipschitz.*

Remark 4 *Assumptions (19)-(21) ensure that the origin is an equilibrium. This assumption can be dropped if the control objective is tracking instead of stabilization of an equilibrium.*

2.2 Control objective

The control objective is to drive the system to the origin in finite time. Designs are discussed to reach the origin in minimum time or, alternatively, to achieve finite-time convergence while preserving smoothness of the closed-loop solution. The control design method can also be applied to solve a class of tracking problems at $x = 0$; see also Remark 16. Moreover, we consider the observer design problem of estimating the distributed state from boundary measurements

$$Y(t) = u(1,t) \tag{22}$$

and the design of an output-feedback controller using measurements $Y(t)$ only.

Remark 5 *Note that in Equation (4), all m components of the state v are assumed to be actuated at the boundary $x = 1$, and that in (22) all n components of the state u are assumed to be measured at the same boundary. In practical applications, the transformation required to bring the physical model to form (1)-(6) can mean that most or all physical boundary values at $x = 1$ must be measured, and the physical actuation signal can be a function of both U and the physical measurements. See for instance [25] for a detailed discussion of a particular case with $n = m = 1$.*

3 State feedback control

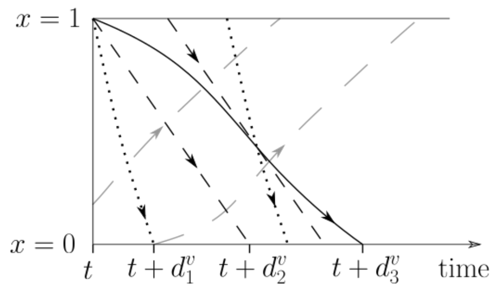

Fig. 1. Characteristic lines of a system with $m = 3$ and $n = 1$. Starting at $x = 1$, the control inputs propagate through the domain along the dotted (U_1), dashed (U_2) and solid (U_3) black lines, respectively. The grey dashed lines indicate the characteristic lines of the state u.

The control design is based on the dynamics on the characteristic lines of system (1)-(6) which are sketched in Figure 1. The control inputs at time t enter the dynamics at $x = 1$ and propagate through the domain with finite speeds λ_i^v. Because the propagation speeds are finite, the control input U_i affects the state v_i only on the characteristic line along which it propagates, but not earlier. In particular, at time t the state $(u,v)(x,t)$ for $x \in [0,1)$ cannot be affected by $U(t)$. Instead, the inputs $U(t)$ are designed to control the state on the characteristic line along which input U_m, which is associated with the slowest transport speed λ_m^v, propagates.

The controller design can be summarized into the following steps:

(1) Derive the dynamics on the characteristic lines along which the inputs propagate through the domain. As shown subsequently, the state on the slowest characteristic line is governed by an ODE coupled to $n+m-1$ PDEs all convecting in positive x-direction, i.e. with inflow boundary at $x=0$, compared to $n+m$ heterodirectional PDEs governing (u,v).
(2) Establish predictability of the states on these characteristic lines using the current state and some boundary values at $x=0$, independently of the control input $U(t)$.
(3) Virtually move the control inputs from $x=1$ to $x=0$, making it straightforward to control the dynamics on the slowest characteristic line and making the prediction operators implementable.
(4) Using the state predictions, construct the input U that ensures that the closed-loop system has the desired boundary value at $x=0$.

3.1 *Dynamics, predictability and boundary values of state on characteristic lines*

Define the characteristic lines along which the inputs propagate as

$$\phi_i^v(x)=\int_x^1 \frac{1}{\lambda_i^v(\xi)}d\xi, \quad x\in[0,1],\ i=1,\dots,m \tag{23}$$

and, to simplify notation later,

$$\phi_0^v(x)=0 \qquad \text{for all } x\in[0,1], \tag{24}$$

and the corresponding delay times

$$d_i^v=\phi_i^v(0). \tag{25}$$

For $j=0,\dots,m$, define the states on the characteristic line along which v_j evolves as

$$\bar{u}_i[j](x,t)=u_i(x,t+\phi_j^v(x)), \tag{26}$$
$$\bar{v}_i[j](x,t)=v_i(x,t+\phi_j^v(x)), \tag{27}$$

with

$$\bar{u}[j](x,t)=\Big(\bar{u}_1[j](x,t)\ \dots\ \bar{u}_n[j](x,t)\Big)^T, \tag{28}$$
$$\bar{v}[j](x,t)=\Big(\bar{v}_1[j](x,t)\ \dots\ \bar{v}_m[j](x,t)\Big)^T. \tag{29}$$

Note that, due to (24),

$$\bar{u}[0](x,t)=u(x,t) \quad \text{and} \quad \bar{v}[0](x,t)=v(x,t). \tag{30}$$

Denote partial derivatives by $\bar{u}_t[j]=\partial_t(\bar{u}[j])$ and $\bar{u}_{i,t}[j]=\partial_t(\bar{u}_i[j])$, and similar for $\bar{v}$ and partial derivatives with respect to x. Assuming for now that the boundary values $v(0,t)$ are available for all t, we introduce the notation

$$w_i[j](t)=v_i(0,t+d_j^v). \tag{31}$$

Note that the boundary values $w_i[j](t)$ at time t depend on the input U_k up to time $t+d_k^v-d_j^v$, i.e. on inputs U_k later than t for $k\le j$ (this will be made more precise later). Finally, define the following state with the $j-th$ component removed:

$$\tilde{v}[j]=\Big(\bar{v}_1[j]\ \dots\ \bar{v}_{j-1}[j]\ \bar{v}_{j+1}[j]\ \dots\ \bar{v}_m[j]\Big) \tag{32}$$

with $\tilde{v}[0]=\bar{v}[0]=v$.

The following lemma characterizes the dynamics of $(\bar{u},\bar{v})$.

Lemma 6 *For $j=0,\dots,m$, the states $(\bar{u}[j],\bar{v}[j])$ satisfy*

$$\bar{u}_t[j](x,t)=-\bar{\Lambda}_j^u\bar{u}_x[j](x,t)+\bar{F}^u[j]((\bar{u},\bar{v})[j](x,t),x,t), \tag{33}$$
$$\tilde{v}_t[j](x,t)=\tilde{\Lambda}_j^v\tilde{v}_x[j](x,t)+\tilde{F}^v[j]((\bar{u},\bar{v})[j](x,t),x,t), \tag{34}$$
$$\bar{v}_{j,x}[j](x,t)=-\bar{F}_j^v[j]((\bar{u},\bar{v})[j](x,t),x,t), \tag{35}$$

with boundary conditions

$$\bar{u}[j](0,t)=f(\bar{v}[j](0,t),t+\phi_j^v(0)), \tag{36}$$
$$\bar{v}_i[j](0,t)=w_i[j](t), \qquad i=1,\dots,j, \tag{37}$$
$$\bar{v}_i[j](1,t)=U_i(t), \qquad i=j+1,\dots,m, \tag{38}$$

and initial conditions

$$\bar{u}_i[j](x,0)=u_i(x,\phi_j^v(x)), \quad i=1,\dots,n, \tag{39}$$
$$\bar{v}_i[j](x,0)=v_i(x,\phi_j^v(x)), \quad i=1,\dots,m,\ i\neq j, \tag{40}$$

where, if $j\ge 1$,

$$\bar{\Lambda}_j^u(x)=\operatorname{diag}\Big(\bar{\lambda}_{j,1}^u(x)\ \dots\ \bar{\lambda}_{j,n}^u(x)\Big), \tag{41}$$
$$\tilde{\Lambda}_j^v(x)=\operatorname{diag}\Big(\bar{\lambda}_{j,1}^v\ \dots\bar{\lambda}_{j,j-1}^v\ \bar{\lambda}_{j,j+1}^v\ \dots\ \bar{\lambda}_{j,m}^v\Big), \tag{42}$$

with

$$\bar{\lambda}_{j,i}^u(x)=\frac{\lambda_i^u(x)\lambda_j^v(x)}{\lambda_i^u(x)+\lambda_j^v(x)}, \tag{43}$$
$$\bar{\lambda}_{j,i}^v(x)=\frac{\lambda_i^v(x)\lambda_j^v(x)}{\lambda_j^v(x)-\lambda_i^v(x)}, \qquad i\neq j, \tag{44}$$

and

$$\bar{F}^u[j] = \left(\bar{F}_1^u[j] \ \dots \ \bar{F}_n^u[j]\right)^T, \tag{45}$$

$$\tilde{F}^v[j] = \left(\bar{F}_1^v[j] \ \dots \ \bar{F}_{j-1}^v[j] \ \bar{F}_{j+1}^v[j] \ \dots \ \bar{F}_m^v[j]\right)^T, \tag{46}$$

with

$$\begin{aligned} \bar{F}_i^u[j] \ ((y,z),x,t) &= \tfrac{\lambda_j^v(x)}{\lambda_i^u(x)+\lambda_j^v(x)} \\ &\times F_i^u((y,z),x,t+\phi_j^v(x)) \end{aligned}, \tag{47}$$

$$\begin{aligned} \bar{F}_i^v[j] \ ((y,z),x,t) &= \tfrac{\lambda_j^v(x)}{\lambda_j^v(x)-\lambda_i^v(x)} \\ &\times F_i^v((y,z),x,t+\phi_j^v(x)) \end{aligned}, \qquad i \neq j, \tag{48}$$

$$\begin{aligned} \bar{F}_j^v[j] \ ((y,z),x,t) &= \tfrac{1}{\lambda_j^v(x)} \\ &\times F_j^v((y,z),x,t+\phi_j^v(x)) \end{aligned}, \tag{49}$$

and, for $j = 0$,

$$\bar{\Lambda}_0^u = \Lambda^u, \quad \bar{\Lambda}_0^v = \Lambda^v, \quad \bar{F}^u[0] = F^u, \quad \bar{F}^v[0] = F^v. \tag{50}$$

PROOF. There is nothing to be shown for $j = 0$ because (33)-(40) with (50) is just a copy of (1)-(6). For $j = 1, \dots, m$, we have the following.

For $i = 1, \dots, m$, we have

$$\bar{v}_{i,t}[j](x,t) = \frac{d}{dt} v_i(x, t + \phi_j^v(x)) = v_{i,t}(x, t + \phi_j^v(x)) \tag{51}$$

(note that $_x$ and $_t$ denote partial derivatives of the state with respect to the space and time, whereas $\frac{d}{dt}$ and $\frac{d}{dx}$ are total derivatives with respect to x and t), and

$$\begin{aligned} \bar{v}_{i,x}[j](x,t) &= \frac{d}{dx} v_i(x, t+\phi_j^v(x)) \\ &= v_{i,x}(x, t+\phi_j^v(x)) - \frac{1}{\lambda_j^v(x)} v_{i,t}(x, t+\phi_j^v(x)) \\ &= \frac{1}{\lambda_i^v(x)} \left[v_{i,t}(x, t+\phi_j^v(x)) \right. \\ &\qquad \left. - F_i^v \left((u,v)(x, t+\phi_j^v(x)), x, t+\phi_j^v(x) \right) \right] \\ &\quad - \frac{1}{\lambda_j^v(x)} v_{i,t}(x, t+\phi_j^v(x)) \\ &= \frac{-\lambda_i^v(x) + \lambda_j^v(x)}{\lambda_i^v(x)\lambda_j^v(x)} \bar{v}_{i,t}(x,t) \\ &\quad - \frac{1}{\lambda_i^v(x)} F_i^v \left((\bar{u},\bar{v})(x,t), x, t+\phi_j^v(x) \right), \end{aligned} \tag{52}$$

where the dynamics (2) were inserted into the third equality of (52). If $i = j$ ($\lambda_i^v = \lambda_j^v$), (52) simplifies to the ODE (35) with (49). Otherwise, substituting (51) into (52) and rearranging gives (34) with (44) and (48).

Repeating the same steps for $\bar{u}_i[j]$ gives (33) with (43) and (47). Here, no case distinction for $i = j$ is required because of the $+$ in the denominator of (43).

Regarding the boundary conditions, note that

$$\frac{\lambda_i^v \lambda_j^v}{\lambda_j^v - \lambda_i^v} > 0 \Leftrightarrow \lambda_j^v > \lambda_i^v \Leftrightarrow j < i. \tag{53}$$

Therefore, for $i < j$ the propagation direction of state $\bar{v}_i[j]$ is in the positive x-direction. Consequently, the boundary condition must be specified on the inflow boundary at $x = 0$ as given in (37) (where definitions (25), (27) and (31) are used) , while for $i > j$, the direction of propagation of $\bar{v}_i[j]$ remains in negative x-direction with inflow boundary at $x = 1$ as given by (38) (using definitions (23), (27) and Equation (4)). The inflow boundary of $\bar{u}[j]$ remains at $x = 0$ for all i and j as given in (36). Since $\bar{v}_j[j]$ satisfies (35), which is an ODE in space without time-dynamics, the boundary value can be specified at any $x \in [0,1]$ (even in the interior of the domain). It is here included in (37). ■

Remark 7 *Note that for all j*

$$\bar{\lambda}_{j,j-1}^v < \dots < \bar{\lambda}_{j,1}^v < 0 < \bar{\lambda}_{j,m}^v < \dots < \bar{\lambda}_{j,j+1}^v, \tag{54}$$

which is straightforward to show using (15). Therefore, the propagation direction of $\bar{v}_i[j]$ remains in the negative x-direction for $i = j+1, \dots, m$, but is in the positive x-direction for $i = 1, \dots, j-1$. Moreover, $\bar{v}_j[j]$ satisfies an ODE. Furthermore, the input associated with the fastest speed in (33)-(38) is U_{j+1}. Also note that F_i^v, F_i^u and f are evaluated at time $t + \phi_j^v(x)$ in (47)-(49) and at time $t + \phi_j^v(0)$ in (36), respectively.

Next, we establish a precise characterization of the dependence and independence of the states $(\bar{u}[j], \bar{v}[j])$ on the control inputs $U(t)$. As we are only interested in dependence on $U(t)$, for now we continue to treat the boundary value at $x = 0$, i.e. $\bar{v}_i[j](0,t)$ for $i = 1, \dots, j$ in (37), as known.

For time $t \geq 0$ and $j = 0, \dots, m$, define the interval

$$\mathcal{I}_j^t = [t, t + (d_{j+1}^v - d_j^v)] \tag{55}$$

and the set of boundary values (which is empty for $j = 0$)

$$\mathbb{W}_j(t) = \left\{ \bar{v}_1[j](0, \mathcal{I}_j^t), \dots, \bar{v}_j[j](0, \mathcal{I}_j^t) \right\}. \tag{56}$$

At time t, $\mathbb{W}_j(t)$ contains the boundary values $\bar{v}_i[j](0, \cdot)$, $i = i, \dots, j$, over the future interval $\mathcal{I}_j^t$.

Lemma 8 *For all $j = 0, \dots, m-1$ and $t \geq 0$ there exists a Lipschitz-continuous operator*

$$\Phi_j^t : \mathcal{X}^n_{(0,1]} \times \mathcal{X}^m_{[0,1)} \times \mathcal{X}^j_{\mathcal{I}^t_j} \to \mathcal{X}^n_{(0,1]} \times \mathcal{X}^m_{[0,1)}$$
$$((\bar{u}, \tilde{v})\,[j](\cdot, t), \mathbb{W}_j(t)) \mapsto (\bar{u}[j+1], \tilde{v})\,[j+1](\cdot, t). \tag{57}$$

Evaluating Φ_j^t is independent of $U(t)$.

PROOF. The proof is based on technical results provided in the appendix. For $j = 0, \dots, m-1$, the system (33)-(40), is of the form (A.1)-(A.7) with $\alpha = \bar{u}[j]$, $\beta = \begin{pmatrix}\bar{v}_1[j] & \dots & \bar{v}_{j-1}[j]\end{pmatrix}$, $\gamma = \bar{v}_j[j]$, $\delta = \begin{pmatrix}\bar{v}_{j+1}[j] & \dots & \bar{v}_m[j]\end{pmatrix}$ and the input arguments of Φ_j^t as the initial condition for the $(\alpha, \beta, \gamma, \delta)$-system. The speed λ_1^δ corresponds to $\bar{\lambda}^v_{j,j+1}$ as per (44). Using ϕ_1^δ as in (A.13), we establish the following relations:

$$\phi_j^v(x) + \phi_1^\delta(x) = \int_x^1 \frac{1}{\lambda_j^v(\xi)} d\xi + \int_x^1 \frac{1}{\bar{\lambda}^v_{j,j+1}(\xi)} d\xi$$
$$= \int_x^1 \frac{1}{\lambda_j^v(\xi)} + \frac{\lambda_j^v(\xi) - \lambda_{j+1}^v(\xi)}{\lambda_{j+1}^v(\xi)\lambda_j^v(\xi)} d\xi = \int_x^1 \frac{1}{\lambda_{j+1}^v(\xi)} d\xi \tag{58}$$

and therefore

$$\bar{u}_i[j]\left(x, t + \phi_1^\delta(x)\right) = u\left(x, t + \phi_j^v(x) + \phi_1^\delta(x)\right)$$
$$= \bar{u}_i[j+1](x,t), \tag{59}$$
$$\bar{v}_i[j]\left(x, t + \phi_1^\delta(x)\right) = \bar{v}_i[j+1](x,t). \tag{60}$$

By Lemma 26 (after shifting time), it is possible to predict

$$\bar{u}[j](x, t + \phi_1^\delta(x)), \qquad x \in (0,1], \tag{61}$$

(corresponding to $\alpha(x, \phi_1^\delta(x))$), and

$$\begin{pmatrix}\bar{v}_1[j] & \dots & \bar{v}_j[j] & \bar{v}_{j+2}[j] & \dots & \bar{v}_m[j]\end{pmatrix}(x, t + \phi_1^\delta(x)), \tag{62}$$

$x \in [0,1)$, (corresponding to β, γ and $(\delta_2 \, \dots \, \delta_{m-j})^T$ at $(x, \phi_1^\delta(x))$) independently of the control inputs $U(t)$. In view of (59)-(60), these are the output arguments of Φ_j^t. Moreover, by Theorem 25, the predicted solution depends Lipschitz-continuously on the input arguments of Φ_j^t. ∎

Remark 9 *The effect of $U_i(t)$ for $i = 1, \dots, j$ on $(\bar{u}, \tilde{v})\,[j+1](\cdot, t)$ is implicitly contained in the boundary values $\mathbb{W}_j(t)$. However, as stated by Lemma 8, evaluating Φ_j^t does not require knowledge of $U(t)$ if $\mathbb{W}_j(t)$ is available.*

Remark 10 (Implementation of Φ_j^t) *The operator Φ_j^t can be implemented by solving the PDE-ODE system (33)-(38) in the rectangular domain*

$$\left\{(x,s) : x \in [0,1], \; s \in [t, t + \int_x^1 \frac{1}{\bar{\lambda}^v_{j,j+1}(\xi)} d\xi]\right\} \tag{63}$$

with $(\bar{u}, \tilde{v})[j]$ as the initial condition and some arbitrarily chosen value for $U(t)$, as this only affects $\bar{v}_{j+1}[j+1]$ but not $\tilde{v}[j+1]$ and $\bar{u}[j+1]$; see the proof of Lemma 26. The parameter t in Φ_j^t is required because the coupling terms F^u and F^v as well as f are allowed to be time-varying.

Finally, we establish the following relationship between the control input U_j, entering at $x = 1$, and the boundary value of v_j at $x = 0$.

Lemma 11 *For given $j \in \{1 \dots m\}$ and $t \geq 0$, consider the Lipschitz-continuous operator defined by*

$$\Psi_j^t : \; \mathcal{X}^n_{(0,1)} \times \mathcal{X}^{m-1}_{(0,1)} \times \mathbb{R} \to \mathbb{R}$$
$$(\phi, \tilde{\varphi}, w) \mapsto \varphi_j(1) \tag{64}$$

where φ_j is the solution of the ODE

$$\varphi_{j,x}(x,t) = -\bar{F}_j^v[j]((\phi, \varphi)[j](x,t), x, t),$$
$$\varphi_j(0) = w, \tag{65}$$

with (analogously to the definition of $\bar{u}$, $\bar{v}$ and $\tilde{v}$)

$$\phi = \begin{pmatrix}\phi_1 & \dots & \phi_n\end{pmatrix} \in \mathcal{X}^n_{(0,1)}, \tag{66}$$
$$\varphi = \begin{pmatrix}\varphi_1 & \dots & \varphi_m\end{pmatrix} \in \mathcal{X}^m_{(0,1)}, \tag{67}$$
$$\tilde{\varphi} = \begin{pmatrix}\varphi_1 & \dots & \varphi_{j-1} & \varphi_{j+1} & \dots & \varphi_m\end{pmatrix} \in \mathcal{X}^{m-1}_{(0,1)}. \tag{68}$$

The component of the state $\bar{v}_j[j]$ as governed by (35) satisfies

$$\bar{v}_j(0,t) = w \tag{69}$$

if and only if

$$U_j(t) = \Psi_j^t\left((\bar{u}, \tilde{v})[j]((0,1), t), w\right). \tag{70}$$

PROOF. As, after a change of notation, (65) is a copy of (35), this follows directly from uniqueness of the solution of the ODE. Existence, uniqueness and Lipschitz-continuous dependence of the solution under the given assumptions is ensured by the Carathéodory existence theorem, see e.g. [12]. ∎

3.2 Target system with virtual control inputs at $x = 0$

Note that for $j = m$, the direction of propagation of all states in (33)-(35) is in the positive x-direction (except for $\bar{v}_m[m]$, which satisfies an ODE in x and therefore has no direction of propagation), and that all boundary conditions are specified at $x = 0$. The system $(\bar{u}[m], \bar{v}[m])$ turns out to be easier to control via the boundary condition at $x = 0$ than it is to control the original system (u, v) via the boundary condition at $x = 1$. Therefore, we first construct a target system for $(\bar{u}[m], \bar{v}[m])$ where we replace the boundary values w in (37) by a new input by virtually moving the control input from $x = 1$ to $x = 0$. Then, in the following section, we construct the control inputs $U(t)$ such that $(\bar{u}[m], \bar{v}[m])$ as given by transformation (26)-(27), with (u, v) governed by (1)-(4) in closed loop with the control law for $U(t)$, is equal to the target dynamics.

We introduce the virtual control inputs $U^*(t) = (U_1^*(t) \dots U_m^*(t))$, which are the desired values for $v(0, t)$. That is, the desired closed-loop trajectories shall satisfy

$$v_i(0, t + d_i^v) = U_i^*(t) \tag{71}$$

for $i = 1, \dots, m$. The shift by d_i^v is to compensate for the transport delay in the system, such that the input $U_i(t)$ at time t is correlated to the virtual input $U_i^*(t)$ at time t. Moreover, we define

$$U_i^*[j](t) = U_i^*(t + d_j^v - d_i^v), \tag{72}$$

which is just another time-shifting such that

$$\bar{v}_i[j](0, t) = U_i^*[j](t). \tag{73}$$

The target system for $(\bar{u}[m], \bar{v}[m])$ in closed loop, which we denote by $(\bar{u}^*, \bar{v}^*)$, is

$$\bar{u}_t^*(x, t) = -\bar{\Lambda}_m^u \bar{u}_x^*(x, t) + \bar{F}^u[m]((\bar{u}^*, \bar{v}^*)(x, t), x, t), \tag{74}$$

$$\tilde{v}_t^*(x, t) = \tilde{\Lambda}_m^v \tilde{v}_x^*(x, t) + \tilde{F}^v[m]((\bar{u}^*, \bar{v}^*)(x, t), x, t), \tag{75}$$

$$\bar{v}_{m,x}^*(x, t) = -\bar{F}_m^v[m]((\bar{u}^*, \bar{v}^*)(x, t), x, t), \tag{76}$$

with boundary conditions

$$\bar{u}^*(0, t) = f(\bar{v}^*(0, t), t + \phi_j^v(0)), \tag{77}$$

$$\bar{v}^*(0, t) = U^*[m](t), \tag{78}$$

and initial conditions

$$\bar{u}_i^*(x, 0) = u_i(x, \phi_m^v(x)), \qquad i = 1, \dots, n, \tag{79}$$

$$\tilde{v}_i^*(x, 0) = v_i(x, \phi_m^v(x)), \qquad i = 1, \dots, m - 1. \tag{80}$$

Note that the equations governing (74)-(77) are equivalent to (33)-(36) for $j = m$ as they cannot be affected by control. The difference is the boundary condition for $v(0, t)$, which is considered as an input in (78) compared to a boundary value that is the result of inputs $U(t) = v(1, t)$ in (37).

It turns out that the choice $U^* = 0$ stabilizes the target system at the origin.

Theorem 12 *Consider system (74)-(78). If*

$$U^*[m](t) = 0 \qquad \text{for all } t \geq t_0 \tag{81}$$

for some $t_0 \geq 0$ (see also Remark 15), then

$$\bar{u}^*(x, t) \equiv \bar{v}^*(x, t) \equiv 0 \qquad \begin{array}{l} \text{for all } x \in [0, 1], \\ t \geq t_0 + \int_0^x \frac{1}{\bar{\lambda}_{m,n}^u(\xi)} d\xi \end{array}, \tag{82}$$

irrespective of the initial condition. Moreover, the closed-loop solution remains bounded for $t < t_0 + \int_0^x \frac{1}{\bar{\lambda}_{m,n}^u(\xi)} d\xi$ if U^* is bounded.

PROOF. System (74)-(78) is of the form (A.1)-(A.7) for $\alpha = \bar{u}^*$, $\beta = \tilde{v}^*$, $\gamma = \bar{v}_n^*$ and $n_\delta = 0$. Due to

$$-\bar{\lambda}_{m,m-1}^v = \frac{\lambda_m^v}{1 - \frac{\lambda_m^v}{\lambda_{m-1}^v}} > \lambda_m^v > \frac{\lambda_m^v}{1 + \frac{\lambda_m^v}{\lambda_n^u}} = \bar{\lambda}_{m,n}^u, \tag{83}$$

the assumptions of Lemma 27 are satisfied. Shifting time and applying the lemma establishes the first part of the result. Boundedness for $t < t_0 + \int_0^x \frac{1}{\bar{\lambda}_{m,n}^u(\xi)} d\xi$ follows directly from Theorem 25. ■

Remark 13 (Exponential stability) *If there exists a constant c such that*

$$\sup_{t \leq t_0} \|U^*[m](t)\| \leq c \| (\bar{u}^*, \bar{v}^*) (\cdot, 0) \|_\infty \tag{84}$$

with $U^*[m](t) = 0 \; \forall t \geq t_0$, then (74)-(78) is also exponentially stable in the sense of that for every $b > 0$ there exist an $a > 0$ such that

$$\| (\bar{u}^*, \bar{v}^*) (\cdot, t) \|_\infty \leq a \| (\bar{u}^*, \bar{v}^*) (\cdot, 0) \|_\infty e^{-bt}. \tag{85}$$

This can also be proven by use of Theorem 25 because, for $t < t_0 + d_m^v$, bound (A.29) and (84) imply the existence of constant a such that

$$e^{b(t_0 + d_m^v)} \| (\bar{u}^*, \bar{v}^*) (\cdot, t) \|_\infty \leq a \| (\bar{u}^*, \bar{v}^*) (\cdot, 0) \|_\infty, \tag{86}$$

while for $t \geq t_0 + d_m^v$ (85) trivially holds due to (82).

3.3 Construction of control inputs

In this section we construct the control inputs that map the open-loop dynamics (1)-(4) into the closed-loop target dynamics (74)-(78) for chosen virtual control inputs U^*. The design exploits the predictability of the state

$(\bar{u}[j], \tilde{v}[j])$ by successively applying the prediction operators Φ_j^t as given in Lemma 8, and then applying the operator Ψ_j^t from Lemma 11 to compute the input U_j that ensures that $\bar{v}[j]$ has the desired boundary value at $x = 0$ as given in (78).

As written in Lemma 8, evaluating Φ_j^t requires knowledge of the boundary values collected in the set $\mathbb{W}_j(t)$ (see definition (56)), which depend on the control inputs U_i over the time interval $[t, t + d_j^v - d_i^v]$, i.e. on future values of U_i for $i \le j$ that are yet to be computed. The crux of making the predictors and control law implementable, is to use the fact that in closed loop these boundary values are equal to the virtual control inputs as in (71) and (73).

For chosen virtual control inputs U^* as in (71)-(73) and using $\mathcal{I}_j^t$ as defined in (55), define the set

$$\mathbb{U}^*[j](t) = \left\{ U_1^*[j] \left(\mathcal{I}_j^t \right), \ldots, U_j^*[j] \left(\mathcal{I}_j^t \right) \right\}, \tag{87}$$

which is the closed-loop analogue of $\mathbb{W}[j]$. Due to (55) and (71)-(72), note that $\mathbb{U}^*[m](t)$ includes future values of U_i^* up to time $t + d_m^v - d_i^v$, which must be predetermined at time t. Using (87), the operators Φ_j^t from Lemma 8 can be applied to successively predict the closed-loop states $(\bar{u}, \tilde{v})[j]$ for $j = 0, \ldots, m-1$ via

$$\begin{aligned} &(\bar{u}[j+1]((0,1], t), \tilde{v}[j+1]([0,1), t)) \\ &\quad = \Phi_j^t \left((\bar{u}[j]((0,1], t), \tilde{v}[j]([0,1), t)), \mathbb{U}_j^*(t) \right) \end{aligned} \tag{88}$$

with $(\bar{u}, \tilde{v})\,[0]$ as in (30). By Lemma 11, the control inputs $U_j(t)$, $j = 1, \ldots, m$ that ensure (73) are

$$U_j(t) = \Psi_j^t(\bar{u}[j], \tilde{v}[j], U_j^*[j](t)). \tag{89}$$

In summary, the control algorithm is as follows.

Algorithm 1 State-feedback control algorithm

Input: time t, state $(u, v)\,(\cdot, t)$,
virtual control inputs $U_i^*([t, t + (d_m^v - d_i^v)])$, $i = 1 \ldots m$

Output: control input $U(t)$

1: set $(\bar{u}, \tilde{v})\,[0](\cdot, t) = (u, v)\,(\cdot, t)$
2: **for** $j = 0, \ldots, m-1$ **do**
3: set $\mathbb{U}^*[j](t)$ as in (87)
4: predict $(\bar{u}, \tilde{v})\,[j+1](\cdot, t)$ using (88)
5: compute $U_{j+1}(t)$ using (89)
6: **end for**

We are now in position to formulate the main theorem on state-feedback control.

Theorem 14 *The system consisting of (1)-(4) in closed loop with $U(t)$ constructed as in Algorithm 1 with $U^*(t) = 0$ for all $t \ge t_0$ satisfies $u(x,t) = v(x,t) = 0$ for all $t \ge t_0 + d_m^v + d_n^u$, where $d_n^u = \int_0^1 \frac{1}{\lambda_n^u(\xi)} d\xi$. Moreover, the closed-loop solution remains bounded for $t < t_0 + d_m^v + d_n^u$ if $U^*(t)$ is bounded for $t < t_0$.*

PROOF. Due to Lemmas 8 and 11, system (1)-(6) in closed loop with $U(t)$ as constructed in Algorithm 1 satisfies

$$u(x,t) = \bar{u}^*(x, t - \phi_m^v(x)), \quad v(x,t) = \bar{v}^*(x, t - \phi_m^v(x)), \tag{90}$$

for all $t \ge \phi_m^v(x)$. Using

$$\begin{aligned} &u(x, t_0 + d_m^v + d_n^u) \\ &\quad = u\left(x, t_0 + \int_0^1 \frac{1}{\lambda_n^u(\xi)} d\xi + \int_0^x \frac{1}{\lambda_m^v(\xi)} d\xi \right) \\ &\quad = \bar{u}^* \left(x, t_0 + \int_x^1 \frac{1}{\lambda_n^u(\xi)} d\xi + \int_0^x \frac{1}{\bar{\lambda}_{m,n}^u(\xi)} d\xi \right) \end{aligned} \tag{91}$$

and analogously for v, applying Theorem 12 proves convergence to the origin, i.e., that (91) is equal to zero, as well as boundedness for $x \in [0,1]$, $t \ge \phi_m^v(x)$. It only remains to show boundedness for $t < \phi_m^v(x)$. For $t < \phi_1^v(x)$, boundedness follows directly from Theorem 25. Then, using boundedness of $U_i^*[j]$ for $1 \le j$, one can apply Theorem 25 recursively for $j = 1, \ldots, m-1$ to prove boundedness of $(\bar{u}, \bar{v})[j](x,t)$ for $x \in [0,1]$, $t < \int_x^1 \frac{1}{\lambda_{j,j+1}(x)}$. By (44) and (26)-(27), this is equivalent to boundedness of $u(x,t)$ and $v(x,t)$ in each of the domains $x \in [0,1]$, $t \in [\phi_j^v(x), \phi_{j+1}^v(x)]$, $j = 1, \ldots, m-1$, which finishes the proof. ■

Remark 15 (Design of $U^*(t)$) *Choosing $U^*(t) = 0$ for all $t \ge 0$ is sufficient (but not necessary) to ensure that the system is driven to the origin in minimum time. However, minimum-time control can create discontinuities in the state which are undesirable in some applications.*

For instance, to obtain classical C^1-solutions to (1)-(6), the coefficients Λ^u, Λ^v, F^u, F^v and f, the initial conditions (u_0, v_0) and the control input U must be C^1-functions of their arguments, and the initial conditions and control inputs must also satisfy the C^1-compatibility conditions at $x = 0$ and $x = 1$. See for instance [5, chapter 3] for a more detailed discussion of regularity of the solution. The corresponding compatibility conditions for U^* are

$$U_i^*(0) = \lim_{s \to d_i^v} v_i(0, s), \quad (U_i^*)'(0) = \lim_{s \to d_i^v} v_i'(0, s). \tag{92}$$

Note that the boundary values $v_i(t)$, $t < d_i^v$, are predictable independent of U; i.e., there exist prediction op-

erators Π_j^t *such that*

$$v_j(0,[0,d_j^v)) = \Pi_j^t(v_0,u_0,\bar{\mathbb{U}}_j^*) \tag{93}$$

where

$$\bar{\mathbb{U}}_j^* = \left\{U_1^*([0,d_j^v - d_1^v]),\ldots,U_{j-1}^*([0,d_j^v - d_{j-1}^v])\right\}. \tag{94}$$

Existence of Π_i^t *can be proven by use of Lemma 26.*

One design $U_i^* \in C^1$ *that reaches zero within some finite time* $T > 0$*, where* T *can be chosen to tune the convergence time and gradient of the solution, is*

$$U_i^*(t) = \begin{cases} c_0 + c_1 t + c_2 t^2 + c_3 t^3 & \text{if } t < T \\ 0 & \text{if } t \geq T \end{cases}. \tag{95}$$

The coefficients c_0 *to* c_3 *are determined by condition (92) and the target condition*

$$U_i^*(T) = 0, \quad (U_i^*)'(T) = 0, \tag{96}$$

which can be solved recursively for $i = 1\ldots m$ *by use of (93).*

Remark 16 (Tracking) *Instead of stabilizing the system at the origin, a tracking objective at* $x = 0$ *of the form*

$$v_i(0,t) = h_i(t) \text{ for } t \geq d_i^v \tag{97}$$

can be satisfied simply by choosing

$$U_i^*(t) = h_i(t + d_i^v). \tag{98}$$

For tracking, assumptions (19)-(21) can be dropped as they are only required to ensure that the origin is an equilibrium. Thus, the terms F^u*,* F^v *and* f *are allowed to include time-varying disturbances for which short-term predictions exist. More precisely, it is sufficient if at each time* $t \geq 0$*, the functions* F^u*,* F^v *and* f *are known over the future interval* $[t, t + d_m^v]$*.*

Remark 17 (Sampled-time control algorithm) *Instead of continuously evaluating the control law as described in Algorithm 1, which is practically intractable as the PDEs need to be solved in zero time when evaluating* Φ_j^t*, a sampled-time version can be implemented that at time* t *computes the control input for the interval* $[t, t+\delta]$*. For this purpose, the state* $(\bar{u},\tilde{v})[m](\cdot,t)$*, which in closed loop is equal to the state of the target system* $(\bar{u}^*,\tilde{v}^*)(\cdot,t)$*, can be predicted based on state* $(u,v)(\cdot,t)$ *by successively evaluating the operators* Φ_j^t *for* $j = 0\ldots m$*, as done in Algorithm 1. Then, the trajectory of* $(\bar{u}^*,\bar{v}^*)(\cdot,s)$ *over the interval* $s \in [t, t+\delta]$ *(as opposed to just the time point* t*) can be obtained by solving (74)-(78) over the interval* $[t, t+\delta]$*. The inputs* U *that render the system into the target system are the boundary values of the target system at* $x = 1$*; i.e.,*

$$U_i(s) = \bar{v}_i^*(s), \quad s \in [t, t+\delta], \quad i = 1,\ldots,m. \tag{99}$$

In algorithmic form this reads as follows.

Algorithm 2 Sampled state-feedback control algorithm

Input: time t, state $(u,v)(\cdot,t)$,
virtual control inputs $U_i^*([t, t + (d_m^v - d_i^v) + \delta])$, $i = 1\ldots m$
Output: control input $U([t, t+\delta])$

set $(\bar{u},\tilde{v})[0](\cdot,t) = (u,v)(\cdot,t)$
for $j = 0,\ldots,m-1$ **do**
  set $\mathbb{U}^*[j](t)$ as in (87)
  predict $(\bar{u},\tilde{v})[j+1](\cdot,t)$ using (88)
end for
set $(\bar{u}^*,\tilde{v}^*)(\cdot,t) = (\bar{u},\tilde{v})[m](\cdot,t)$
solve (74)-(78) over time-interval $[t, t+\delta]$
set $U_i(s) = \bar{v}_i^t(s)$ for $s \in [t, t+\delta]$, $i = 1,\ldots,m$

However, this algorithm uses predictions of the closed-loop trajectories over the longer horizon

$$\{(x,s) : x \in [0,1],\ s \in [t, t + \phi_m^v(x) + \delta]\} \tag{100}$$

compared to

$$\{(x,s) : x \in [0,1],\ s \in [t, t + \phi_m^v(x)]\} \tag{101}$$

for Algorithm 1. In theory, if exact predictions are assumed, the control inputs computed via the two algorithms are equivalent. In practice, however, uncertainty will lead to prediction errors that can affect closed loop stability. It is to be expected that longer prediction horizons due to sampling lead to more sensitivity to uncertainty. Yet, if the sampling period δ *is small compared to the prediction horizon* d_m^v *of the continuous-time control law, it might be expected that the error introduced by sampling is small. Robustness analysis of closed-loop stability and performance using both the continuous (Algorithm 1) and sampled (Algorithm 2) feedback laws is the subject of further research.*

Remark 18 *If system (1)-(6) is linear, one would expect that the control input is a linear function of the state and, due to Riesz representation theorem on* $L^2([0,1],\mathbb{R}^{n+m})$*, could be written in the form*

$$U(t) = \int_0^1 K^u(x)u(x,t) + K^v(x)v(x,t)dx \tag{102}$$

with gains $K^u \in L^2([0,1],\mathbb{R}^n)$ *and* $K^v \in L^2([0,1],\mathbb{R}^m)$*. This was done in [23, Section 3.4] for* $n = m = 1$*, in which case the control law turned out to be equivalent* *to the backstepping controller. Although we have so far been unable to derive an expression of form (102) for general n and m, it would be an interesting question to research if it is possible, and to investigate the relation to the different backstepping controllers. In particular, there might be links between degrees of freedom in U^* and degrees of freedom in the target system and kernel boundary conditions of the backstepping method.*

4 Observer design and output feedback

The observer design is based on the observation that the measurements $Y_i(t) = u_i(1,t)$ propagate through the domain along the characteristic lines of the states u_i, similar to how the control inputs U_i propagate along the characteristic lines of states v_i. See also Figure 2. In particular, due to the finite propagation speeds, the state at (x,t) for $x < 1$ has no effect on the measurements at time t. Therefore, the idea behind the observer design is to first estimate the state on the characteristic lines on which the measurements associated with the slowest speed evolved, and then use a prediction operator to construct the estimate of the current state from the estimated state on that characteristic line.

4.1 Dynamics on characteristic lines of measurement

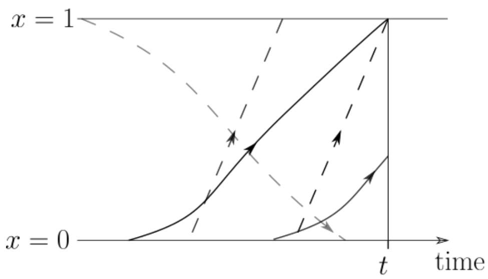

Fig. 2. Characteristic lines of a system with $n = 2$ and $m = 1$ along which the measurements $Y_1(t)$ (solid) and $Y_2(t)$ (dashed) evolve, as well as the characteristic lines of the v.

For $i = 1, \dots, n$ define

$$\phi_i^u(x) = \int_x^1 \frac{1}{\lambda^u(\xi)_i} d\xi, \qquad d_i^u = \phi_i^u(0). \tag{103}$$

Define the states on the characteristic line along which the j-th measurement evolved as

$$\hat{u}_i[j](x,t) = u_i(x, t - \phi_j^u(x)), \tag{104}$$
$$\hat{v}_i[j](x,t) = v_i(x, t - \phi_j^u(x)), \tag{105}$$

with

$$\hat{u}[j](x,t) = \left(\hat{u}_1[j](x,t) \ \dots \ \hat{u}_n[j](x,t)\right), \tag{106}$$
$$\hat{v}[j](x,t) = \left(\hat{v}_1[j](x,t) \ \dots \ \hat{v}_m[j](x,t)\right), \tag{107}$$

and the state with j-th element removed

$$\check{u}[j] = \left(\hat{u}_1[j] \ \dots \ \hat{u}_{j-1}[j] \ \hat{u}_{j+1}[j] \ \hat{u}_n[j]\right)^T. \tag{108}$$

Lemma 19 *For $j = 1, \dots, n$, the states $(\hat{u}[j], \hat{v}[j])$ satisfy*

$$\check{u}_t[j](x,t) = -\check{\Lambda}_j^u \check{u}_x[j](x,t) + \check{F}^u[j]((\hat{u},\hat{v})[j](x,t),x,t), \tag{109}$$
$$\hat{u}_{j,x}[j](x,t) = \hat{F}_j^u[j]((\hat{u},\hat{v})[j](x,t),x,t), \tag{110}$$
$$\hat{v}_t[j](x,t) = \hat{\Lambda}_j^v \hat{v}_x[j](x,t) + \hat{F}^v[j]((\hat{u},\hat{v})[j](x,t),x,t), \tag{111}$$

with boundary conditions

$$\hat{u}_i[j](0,t) = f_i(\hat{v}[j](0,t), t - d_j^u), \quad i = j+1, \dots, n, \tag{112}$$
$$\hat{u}_i[j](1,t) = Y_i(t), \qquad i = 1, \dots, j, \tag{113}$$
$$\hat{v}[j](1,t) = U(t), \tag{114}$$

and "initial" conditions

$$\check{u}[j](x, \phi_j^u(x)) = u_0(x), \tag{115}$$
$$\hat{v}[j](x, \phi_j^u(x)) = v_0(x), \tag{116}$$

where

$$\check{\Lambda}_j^u(x) = \operatorname{diag}\left(\hat{\lambda}_{j,1}^u \dots \hat{\lambda}_{j,j-1}^u \ \hat{\lambda}_{j,j+1}^u \ \dots \ \hat{\lambda}_{j,n}^u\right), \tag{117}$$
$$\hat{\Lambda}_j^v(x) = \operatorname{diag}\left(\hat{\lambda}_{j,1}^v(x) \ \dots \ \hat{\lambda}_{j,n}^v(x)\right), \tag{118}$$

with

$$\hat{\lambda}_{j,i}^u(x) = \frac{\lambda_i^u \lambda_j^u}{\lambda_j^u - \lambda_i^u}, \qquad i \neq j, \tag{119}$$
$$\hat{\lambda}_{j,i}^v(x) = \frac{\lambda_i^v \lambda_j^u}{\lambda_i^v + \lambda_j^u}, \tag{120}$$

and

$$\check{F}^u[j] = \left(\hat{F}_1^u[j] \ \dots \ \hat{F}_{j-1}^u[j] \ \hat{F}_{j+1}^u[j] \ \dots \ \hat{F}_n^u[j]\right)^T, \tag{121}$$
$$\hat{F}^v[j] = \left(\hat{F}_1^v[j] \ \dots \ \hat{F}_m^v[j]\right)^T, \tag{122}$$

with

$$\hat{F}_i^u[j] \ ((y,z),x,t) = \frac{\lambda_j^u(x)}{\lambda_j^u(x) - \lambda_i^u(x)} \times F_i^u((y,z),x,t-\phi_j^u(x)), \quad i \neq j, \tag{123}$$

$$\hat F^u_j[j]\ ((y,z),x,t) = \frac{1}{\lambda^u_j(x)} \times F^u_j((y,z),x,t-\phi^u_j(x)), \tag{124}$$

$$\hat F^v_i[j]\ ((y,z),x,t) = \frac{\lambda^u_j(x)}{\lambda^u_j(x)+\lambda^v_i(x)} \times F^v_i((y,z),x,t-\phi^u_j(x)). \tag{125}$$

PROOF. The proof is very similar to the proof of Lemma 6. ∎

Note that because of (14),

$$\hat\lambda^u_{j,j-1} < \ldots < \hat\lambda^u_{j,1} < 0 < \hat\lambda^u_{j,n} < \ldots < \hat\lambda^u_{j,j+1}. \tag{126}$$

Therefore, the boundary conditions for $\hat u_i[j]$ are specified at the inflow boundary at $x=0$ for $i>j$ and at $x=1$ for $i<j$, while $\hat u_j[j]$ is an ODE in x and therefore has no direction of propagation. The states $(\hat u,\hat v)\,[j]$ are related to the current state (u,v) via a prediction operator that is independent of the input $U(t)$.

Lemma 20 *For $j=1,\ldots,n$ there exists a Lipschitz-continuous operator*

$$\begin{aligned}\Sigma^t_j:\ &\mathcal{X}^{n+m}_{[0,1)} \to \mathcal{X}^{n+m}_{[0,1)}\\ &(\hat u,\hat v)[j](\cdot,t) \mapsto (u,v)\,(\cdot,t).\end{aligned} \tag{127}$$

Evaluating Σ^t_j is independent of $U(t)$.

PROOF. By Lemma 28 (after shifting time), (1)-(4) with initial condition

$$(u,v)(x,t-\phi^u_j(x)) = (\hat u,\hat v)[j](x,t) \tag{128}$$

has a unique solution in the domain

$$\big\{(x,s):\ x\in[0,1),\ s\in[t-\phi^u_j(x),t+\phi^v_1(x))\big\} \tag{129}$$

that is independent of $U(t)$. This domain includes the current state at $([0,1),t)$, which is the output argument of Σ^t_j. Moreover, by Theorem 25, the solution in this domain depends Lipschitz-continuously on the input arguments of Σ^t_j. ∎

4.2 Observer

In order to estimate the states $(\hat u[j],\hat v[j])$ for some j by use of an observer, it is desirable to have the boundary values of all states at the inflow boundary available to the observer, and to have that the inflow boundary (either $x=0$ or $x=1$) is equal for all states, i.e. that all states propagate in the same direction. In (113) the boundary value is the measurement, i.e. known, whereas the boundary value (112) depends on $\hat v(0,t)$, which is not measured and needs to be estimated. Moreover, the direction of propagation of $\hat v[j]$ is always negative (due to positivity of $\hat\lambda^v_{j,i}$ for all i,j), and the boundary value $\hat v(1,t)$, the control input, is known.

Therefore, $j=n$ is the only choice that ensures that the inflow boundary of all states is equal and that all inflow boundary values are known. We design the observer for the $(\hat u[n],\hat v[n])$-state as a copy of (109)-(111) complemented by the boundary values and an initial estimate.

Theorem 21 *Consider the observer*

$$\check u^o_t(x,t) = -\check\Lambda^u_n \check u^o_x(x,t) + \check F^u[n]((\hat u^o,\hat v^o)(x,t),x,t), \tag{130}$$

$$\hat u^o_{n,x}(x,t) = \hat F^u_n[n]((\hat u^o,\hat v^o)(x,t),x,t), \tag{131}$$

$$\hat v^o_t(x,t) = \hat\Lambda^v_n \hat v^o_x(x,t) + \hat F^v[n]((\hat u^o,\hat v^o)(x,t),x,t), \tag{132}$$

$$\hat u^o(1,t) = Y(t), \tag{133}$$

$$\hat v^o(1,t) = U(t), \tag{134}$$

$$\check u^o(x,0) = \check u^o_0(x), \tag{135}$$

$$\hat v^o(x,0) = \hat v^o_0(x). \tag{136}$$

For any initial guess $\check u^o_0$, $\hat v^o_0$, the state-estimates satisfy

$$\hat u^o(x,t) = \hat u[n](x,t), \qquad \hat v^o(x,t) = \hat v[n](x,t), \tag{137}$$

for all $x\in[0,1]$, $t \ge \int_x^1 \frac{1}{\hat\lambda^v_{n,m}(\xi)}d\xi$.

PROOF. The estimation error

$$\hat e^u(x,t) = \hat u^o(x,t) - \hat u[n](x,t), \tag{138}$$

$$\hat e^v(x,t) = \hat v^o(x,t) - \hat v[n](x,t), \tag{139}$$

satisfies (subtracting (109)-(114) from (130)-(134))

$$\check e^u_t(x,t) = -\check\Lambda^u_n \check e^u_x(x,t) + E^u(\hat u[n],\hat v[n],\hat u^o,\hat v^o,x,t), \tag{140}$$

$$\hat e^u_{n,x}(x,t) = E^u_n(\hat u[n],\hat v[n],\hat u^o,\hat v^o,x,t), \tag{141}$$

$$\hat e^v_t(x,t) = \hat\Lambda^v_n \hat e^v_x(x,t) + E^v(\hat u[n],\hat v[n],\hat u^o,\hat v^o,x,t), \tag{142}$$

$$e^u(1,t) = 0, \tag{143}$$

$$e^v(1,t) = 0, \tag{144}$$

with

$$\begin{aligned}E^u(\hat u[n],\hat v[n],\hat u^o,\hat v^o,x,t) = {}&\hat F^u[n]((\hat u^o,\hat v^o)(x,t),x,t)\\ &- \hat F^u[n]((\hat u[n],\hat v[n])(x,t),x,t),\end{aligned} \tag{145}$$

$$\begin{aligned}E^v(\hat u[n],\hat v[n],\hat u^o,\hat v^o,x,t) = {}&\hat F^u[n]((\hat u^o,\hat v^o)(x,t),x,t)\\ &- \hat F^u[n]((\hat u[n],\hat v[n])(x,t),x,t).\end{aligned} \tag{146}$$

Note that

$$\hat{e}^u(x,t) = \hat{e}^v(x,t) = 0, \tag{147}$$

i.e. $\hat{u}^o(x,t) = \hat{u}[n](x,t)$ and $\hat{v}^o(x,t) = \hat{v}[n](x,t)$, implies

$$E^u(\hat{u}[n], \hat{v}[n], \hat{u}^o, \hat{v}^o, x, t) = 0, \tag{148}$$
$$E^v(\hat{u}[n], \hat{v}[n], \hat{u}^o, \hat{v}^o, x, t) = 0. \tag{149}$$

After a change of coordinates from x to $1-x$ and ordering the states according to speeds, the error dynamics (140)-(144) has the form (A.1)-(A.7) and satisfies all assumptions of Lemma 27 with slowest speed $\hat{\lambda}^v_{n,m}$. Therefore, applying Lemma 27 completes the proof. ■

Combining Lemma 20 and Theorem 21, we obtain the main result on state estimation.

Theorem 22 *The observer consisting of (130)-(134) with*

$$(u_{est}, v_{est})(\cdot, t) = \Sigma^t_n\left((\hat{u}^o, \hat{v}^o)(\cdot, t)\right) \tag{150}$$

achieves exact state-estimation within $d^v_m + d^u_n$, i.e.

$$(u_{est}, v_{est})(x,t) = (u,v)(x,t) \tag{151}$$

for all $x \in [0,1]$, $t \geq d^v_m + d^u_n$.

Note that evaluating Σ^t_n does not depend on $U(t)$. $U(t)$ is only needed to update the observer (130)-(134) as time proceeds.

Remark 23 (Anti-collocated observer) *It is straightforward to design an observer that uses measurements that are anti-collocated with the control input, i.e.*

$$Y_1(t) = u(0,t), \qquad Y_2(t) = v(0,t), \tag{152}$$

by making a change of coordinates from x to $1-x$, switching the roles of u and v, and repeating the previous steps. Similar to (109)-(114), the states

$$\acute{u} = u\left(x, t - \int_0^x \frac{1}{\lambda^v_m(\xi)} d\xi\right), \tag{153}$$
$$\acute{v} = v\left(x, t - \int_0^x \frac{1}{\lambda^v_m(\xi)} d\xi\right), \tag{154}$$

all evolve in positive x-direction, and the inflow boundary at $x=0$ as given in (152) is available to the observer. Therefore, $(\acute{u}, \acute{v})$ can be estimated in finite time, and the current state can be determined by use of a prediction operator similar to Σ^t_j.

4.3 Output feedback control

The output-feedback control problem is solved by combining the observer from Section 4.2 with the state-feedback controller from Section 3.3.

Theorem 24 *Consider the output feedback controller consisting of the observer (130)-(134) with the state-estimate as in (150) and $U(t)$ as constructed in Algorithm 1 with $U^*[m](t) = 0$ for all $t \geq t_0$. For any initial guess $(\check{u}^o_0, \hat{v}^o_0)$ and all initial conditions (u_0, v_0), the closed-loop solution satisfies $u(x,t) = v(x,t) = 0$ for all $t \geq \max\{t_0 + d^v_m + d^u_n, 2(d^v_m + d^u_n)\}$, and remains bounded for $t < \max\{t_0 + d^v_m + d^u_n, 2(d^v_m + d^u_n)\}$ if $U^*(t)$ is bounded.*

PROOF. By Theorem 22, the estimation errors become zero after $t \geq d^v_m + d^u_n$. Once both the estimation errors and $U^*[m]$ are zero, whichever occurs last, by Theorem 14 it takes another $d^v_m + d^u_n$ units of time until the states reach the origin.

To prove boundedness of the solution for $t < \max\{t_0 + d^v_m + d^u_n, 2(d^v_m + d^u_n)\}$, one can apply the methodology in the proof of Theorem 25. The overall closed-loop dynamics consisting of the observer dynamics $(\hat{u}^o, \hat{v}^o)$, the state (u,v) as well as the dynamics in the operators Σ^t_n, Φ^t_i and Ψ^t_i, $i = 1, \ldots, n$, have a slightly more complex structure than (A.1)-(A.7) but satisfy the same basic assumptions, namely boundedness of external inputs (only the virtual inputs U^* in this case) and Lipschitz-continuity of state-dependent functions. Therefore, one can transform the overall closed-loop dynamics into integral equations and apply a successive approximation method to prove existence of the solution as well as a bound that grows at most exponentially in time with a rate proportional to the Lipschitz constants. ■

5 Numerical example and implementation

We demonstrate the controller performance in a numerical example with $n = m = 2$ with

$$\lambda^u_1 = 2, \qquad \lambda^u_2 = 1.5, \qquad \lambda^v_1 = 1.5,$$
$$\lambda^v_2 = \begin{cases} 1 & x \geq 0.5 \\ 0.8 & x < 0.5 \end{cases}, \qquad f = \begin{pmatrix} -5\min\{|v_1|, v_1^2\} \\ -v_2 \end{pmatrix},$$
$$F^u = \begin{pmatrix} \sin(u_1 + v_2) \\ \sin(u_1)\cos(u_2) \end{pmatrix}, \quad F^v = \begin{pmatrix} \sin(v_1 - u_1) \\ F^v_2 = (v_2 - u_2) \end{pmatrix}. \tag{155}$$

The resulting delays are $d^u_n = \frac{2}{3}$ and $d^v_m = 1.125$. The initial conditions of all states are set to one and the initial condition of the observer is set to zero. In open loop ($U \equiv 0$), the states u_2 and v_2 diverge.

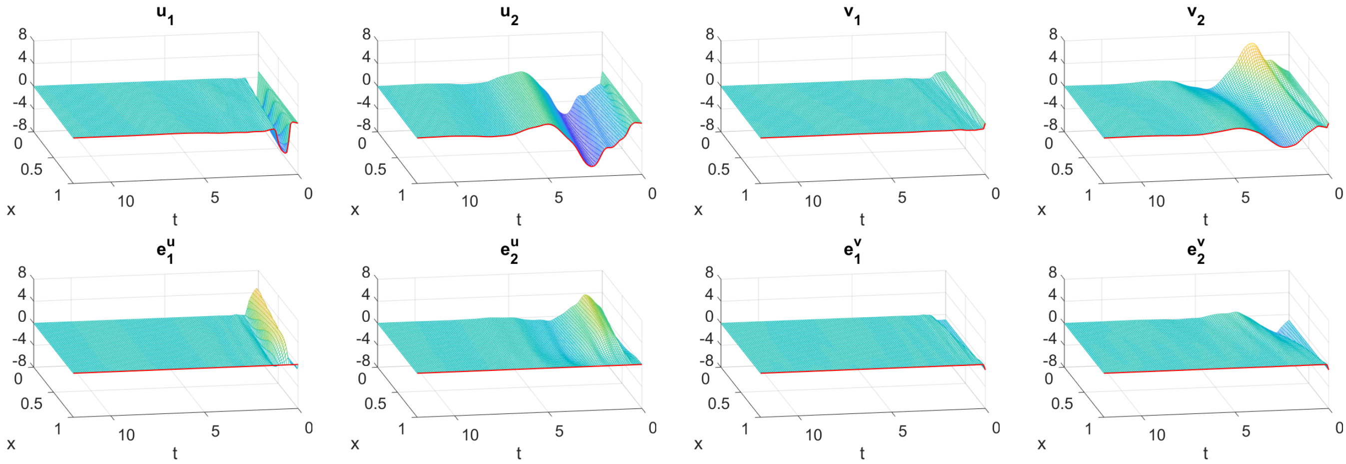

Fig. 3. Closed-loop trajectories (top row) and estimation errors $e^u = u_{est} - u$ and $e^v = v_{est} - v$ with u_{est}, v_{est} as in (150) (bottom row). The red lines indicate the measurement, control input and boundary estimation error, respectively.

The closed-loop trajectories using the output-feedback controller (Algorithm 1 combined with the observer (130)-(134) with (150)) with $U^*(t) = 0$ for all $t \geq 0$, and the estimation errors are depicted in Figure 3. The evolution of the state norm is also shown in Figure 4. The input propagation fronts are clearly visible. In accordance with theory, states and estimation errors become zero in finite time, up to numerical errors. In Figure 4 one can also see the faster convergence time in the state-feedback case.

For the simulations, all PDEs (i.e. the system (1)-(6) and those in the prediction operators Φ_j^t, $j = 0, \ldots, m-1$, and Σ_n^t) are discretized in space using first-order finite differences, leading to a high-order ODE ("method of lines") that is solved by use of Matlab's `ode45`. In the implementation of Φ_j^t, Remark 10 is exploited to avoid having to solve the PDEs over non-rectangular domains. The operator Σ_n^t is implemented similarly over a rectangular domain. In the present implementation, the continuous-time algorithm is used, i.e. the control law is evaluated every time `ode45` evaluates the right-hand side of the discretized PDEs. For a spatial grid with 50 elements, which has been used to produce the figures, the average computation time to evaluate the output feedback controller is about 0.15 s on a standard laptop, although it should be emphasized that the current code has not been optimized for performance. In practical applications the sampled-time Algorithm 2 would be more appropriate, and there seems to be a trade-off between sampling time, which likely affects prediction errors due to model uncertainty, and the number of discretization elements, which affects both numerical prediction errors and the allowable sampling time (via the computation time). All this is the subject of further research. Also note that for most practical systems, a state transformation is required to first bring the model to form (1)-(6) before the control law can be implemented. Then, the measurement $Y(t)$ is obtained from the physical measurements and the state transformation, the control law is implemented in the characteristic form (1)-(6) as outlined in this paper, and the control input $U(t)$ is mapped back into the physical actuation signal by inverting the transformation. See also Remark 5 and [25].

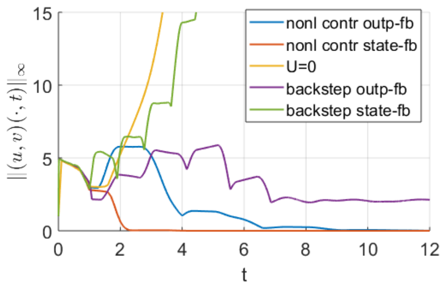

Fig. 4. Comparison of the state norm trajectory using the nonlinear controller presented in this paper to the backstepping controller from [16] for both state and output feedback.

In Figure (4), we compare the performance of our nonlinear controller to the linear backstepping controller from [16]. As the backstepping controller is not well-defined for discontinuous transport speeds and to allow a fairer comparison, we modified λ_2^v to the constant value $\lambda_2^v(x) = \frac{8}{9}$ (for both our nonlinear and the backstepping controller). The backstepping controller is computed by linearizing (155) (with the modified λ_2^v) around the origin. As opposed to the nonlinear controller, the backstepping controller fails to steer the state to the origin. By use of state feedback backstepping, the state diverges only slightly slower than the uncontrolled system ($U = 0$). For this initial condition, the output feedback controller actually performs better than the state-feedback controller and the closed-loop trajectory converges to a nonzero value. By simulating a large sample of initial conditions, it appears that the output feedback backstepping controller can asymptotically stabilise the origin of the nonlinear system if $\|(u_0, v_0)\|_\infty \lesssim 0.07$ (mind numerical inaccuracies and that only a finite number of initial conditions has been tested). In any case, the backstepping controller cannot reliably stabilise the nonlinear system if the initial conditions are far from the origin, whereas the nonlinear controller presented in this paper has no restriction on the initial condition.

6 Conclusions

We solved the problem of stabilizing systems governed by general semilinear hyperbolic PDEs in one dimension with actuation and sensing restricted to one boundary of the domain. The design exploits the dynamics and predictability of the states on the characteristic lines of the system. The result is applicable to relevant practical systems such as multiphase flow or multi-lane traffic.

In the presented design, the virtual inputs U^* are a degree of freedom. Convergence to the origin is guaranteed if the virtual inputs become zero, but the transients before reaching the origin can be tuned via U^*. Therefore, the design of U^* should be investigated further.

Future work will also investigate robustness of the proposed control scheme with respect to model uncertainty, which is of high interest for practical application. One line of current research is trying to prove robustness by exploiting continuous dependence of the solution on the model parameters [5], so that it might be possible to show that the prediction error and the error between desired and actual trajectories is small if appropriate assumptions are made. While simulations suggest a certain degree of robustness at least in some cases, rigorous robustness certificates would be highly desirable. Further, the effect of the prediction horizon in the sampled-time control law (Algorithm 2) on robustness compared to the continuous-time implementation is of interest.

Finally, the method should be extended to different classes of systems such as systems with bilateral actuation and sensing, which should improve performance and robustness compared to the unilateral case, underactuated systems with less inputs than states, networked systems, and quasilinear systems (see [27] for a first result on quasilinear systems).

Acknowledgements

The authors would like to thank Henrik Anfinsen for help with the implementation of the backstepping controller.

References

[1] Ole Morten Aamo. Disturbance rejection in 2×2 linear hyperbolic systems. *IEEE Transactions on Automatic Control*, 58(5):1095–1106, 2013.

[2] Jean Auriol and Florent Di Meglio. Minimum time control of heterodirectional linear coupled hyperbolic PDEs. *Automatica*, 71:300–307, 2016.

[3] A Aw and Michel Rascle. Resurrection of "second order" models of traffic flow. *SIAM Journal on Applied Mathematics*, 60(3):916–938, 2000.

[4] Georges Bastin and Jean-Michel Coron. On boundary feedback stabilization of non-uniform linear $2\times$ 2 hyperbolic systems over a bounded interval. *Systems & Control Letters*, 60(11):900–906, 2011.

[5] Alberto Bressan. *Hyperbolic systems of conservation laws: the one-dimensional Cauchy problem*, volume 20. Oxford University Press, 2000.

[6] Marco Cirina. Boundary controllability of nonlinear hyperbolic systems. *SIAM Journal on Control*, 7(2):198–212, 1969.

[7] Jean-Michel Coron, Brigitte d'Andrea Novel, and Georges Bastin. A strict Lyapunov function for boundary control of hyperbolic systems of conservation laws. *IEEE Transactions on Automatic Control*, 52(1):2–11, 2007.

[8] Jean-Michel Coron, Long Hu, and Guillaume Olive. Finite-time boundary stabilization of general linear hyperbolic balance laws via Fredholm backstepping transformation. *Automatica*, 84:95–100, 2017.

[9] Jean-Michel Coron, Rafael Vazquez, Miroslav Krstic, and Georges Bastin. Local exponential H^2 stabilization of a 2×2 quasilinear hyperbolic system using backstepping. *SIAM Journal on Control and Optimization*, 51(3):2005–2035, 2013.

[10] Jonathan de Halleux, Christophe Prieur, J-M Coron, Brigitte d'Andréa Novel, and Georges Bastin. Boundary feedback control in networks of open channels. *Automatica*, 39(8):1365–1376, 2003.

[11] Florent Di Meglio. *Dynamics and control of slugging in oil production*. PhD thesis, École Nationale Supérieure des Mines de Paris, 2011.

[12] Aleksei Fedorovich Filippov. *Differential equations with discontinuous right-hand side*. Kluwer Academic Publishers, 1988.

[13] James M Greenberg and Li Ta Tsien. The effect of boundary damping for the quasilinear wave equation. *Journal of Differential Equations*, 52(1):66–75, 1984.

[14] Martin Gugat, Michael Herty, and Veronika Schleper. Flow control in gas networks: exact controllability to a given demand. *Mathematical Methods in the Applied Sciences*, 34(7):745–757, 2011.

[15] Martin Gugat and Günter Leugering. Global boundary controllability of the de St. Venant equations between steady states. In *Annales de l'IHP Analyse non linéaire*, volume 20, pages 1–11, 2003.

[16] Long Hu, Florent Di Meglio, Rafael Vazquez, and Miroslav Krstic. Control of homodirectional and general heterodirectional linear coupled hyperbolic PDEs. *IEEE Transactions on Automatic Control*, 61(11):3301–3314, 2016.

[17] Long Hu, Rafael Vazquez, Florent Di Meglio, and Miroslav Krstic. Boundary exponential stabilization of 1-dimensional inhomogeneous quasi-linear hyperbolic systems. *SIAM Journal on Control and Optimization*, 57(2):963–998, 2019.

[18] Ta-Tsien Li and Bo-Peng Rao. Exact boundary controllability for quasi-linear hyperbolic systems. *SIAM Journal on Control and Optimization*, 41(6):1748–1755, 2003.

[19] Tatsien Li. Exact boundary observability for quasilinear hyperbolic systems. *ESAIM: Control, Optimisation and Calculus of Variations*, 14(4):759–766, 2008.

[20] Tatsien Li, Ke Wang, and Qilong Gu. *Exact Boundary Controllability of Nodal Profile for Quasilinear Hyperbolic Systems*. Springer, 2016.

[21] David L Russell. Controllability and stabilizability theory for linear partial differential equations: recent progress and open questions. *Siam Review*, 20(4):639–739, 1978.

[22] Andrey Smyshlyaev and Miroslav Krstic. Closed-form boundary state feedbacks for a class of 1-d partial integro-differential equations. *IEEE Transactions on Automatic Control*, 49(12):2185–2202, 2004.

[23] Timm Strecker and Ole Morten Aamo. Output feedback boundary control of 2 × 2 semilinear hyperbolic systems. *Automatica*, 83:290–302, 2017.

[24] Timm Strecker and Ole Morten Aamo. Output feedback boundary control of series interconnections of 2×2 semilinear hyperbolic systems. In *20th IFAC World Congress, 2017*. IFAC, 2017.

[25] Timm Strecker and Ole Morten Aamo. Rejecting heave-induced pressure oscillations in a semilinear hyperbolic well model. In *American Control Conference (ACC), 2017*, pages 1163–1168. IEEE, 2017.

[26] Timm Strecker and Ole Morten Aamo. Two-sided boundary control and state estimation of 2 × 2 semilinear hyperbolic systems. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 2511–2518. IEEE, 2017.

[27] Timm Strecker, Ole Morten Aamo, and Michael Cantoni. Direct predictive boundary control of a first-order quasilinear hyperbolic PDE. In *2019 IEEE 58th Annual Conference on Decision and Control (CDC)*. IEEE, 2019.

[28] Rafael Vazquez, Miroslav Krstic, and Jean-Michel Coron. Backstepping boundary stabilization and state estimation of a 2× 2 linear hyperbolic system. In *2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pages 4937–4942, 2011.

[29] Enrique Zuazua. Exact controllability for semilinear wave equations in one space dimension. In *Annales de l'IHP Analyse non linéaire*, volume 10, pages 109–129, 1993.

A Transformation to integral equations and technical proofs

In the following we establish several technical results. Due to page restrictions, detailed proofs cannot be shown here but references to similar cases in the literature are given.

The following class of systems is generic enough to contain all systems occurring in this paper except the full closed-loop system in Theorem 24.

$$\alpha_t(x,t) = -\Lambda^\alpha(x)\alpha_x(x,t) + G^\alpha(\eta(x,t),x,t), \quad \text{(A.1)}$$
$$\beta_t(x,t) = -\Lambda^\beta(x)\beta_x(x,t) + G^\beta(\eta(x,t),x,t), \quad \text{(A.2)}$$
$$\gamma_x(x,t) = G^\gamma(\eta(x,t),x,t), \quad \text{(A.3)}$$
$$\delta_t(x,t) = \Lambda^\delta(x)\delta_x(x,t) + G^\delta(\eta(x,t),x,t), \quad \text{(A.4)}$$

where $\eta = \begin{pmatrix}\alpha^T & \beta^T & \gamma^T & \delta^T\end{pmatrix}^T$, with boundary conditions

$$\alpha(0,t) = g(\eta_0(t),t), \qquad \beta(0,t) = V^\beta(t), \quad \text{(A.5)}$$
$$\gamma(0,t) = V^\gamma(t), \qquad \delta(1,t) = V^\delta(t), \quad \text{(A.6)}$$

where $\eta_0(t) = \begin{pmatrix}\beta(0,t) & \gamma(0,t) & \delta(0,t)\end{pmatrix}^T$ and V^β, V^γ and V^δ are inputs, and initial conditions

$$\alpha(x,0) = \alpha_0(x), \quad \beta(x,0) = \beta_0(x), \quad \delta(x,0) = \gamma_0(x). \quad \text{(A.7)}$$

The states are vector-valued with

$$\chi(x,t) = \begin{pmatrix}\chi_1(x,t) & \dots & \chi_{n_\chi}(x,t)\end{pmatrix}^T \quad \text{(A.8)}$$

for $\chi = \alpha,\beta,\gamma,\delta$ with $n_\chi \geq 0$ (this includes the case where one or more of the states $\alpha,\beta,\gamma,\delta$ is not present). The system coefficients are assumed to satisfy

$$\|G^\chi(\zeta_1,x,t) - G^\chi(\zeta_2,x,t)\| \leq L_\chi\|\zeta_1-\zeta_2\|, \quad \text{(A.9)}$$
$$\|g(\zeta_3,t) - g(\zeta_4,t)\| \leq L_g\|\zeta_3-\zeta_4\|, \quad \text{(A.10)}$$

for all $\zeta_1,\zeta_2 \in \mathbb{R}^{n_\alpha+n_\beta+n_\gamma+n_\delta}$, $\zeta_3,\zeta_4 \in \mathbb{R}^{n_\beta+n_\gamma+n_\delta}$, $x \in [0,1]$, $t \geq 0$ and $\chi = \alpha,\beta,\gamma,\delta$,

$$\lambda_1^\chi(x) > \lambda_2^\chi(x) > \dots > \lambda_{n_\chi}^\chi(x) > 0 \quad \text{(A.11)}$$

for all x and $\chi = \alpha,\beta,\delta$, and

$$G_\chi(0,x,t) = 0, \qquad g(0,t) = 0, \quad \text{(A.12)}$$

for all $x \in [0,1]$, $t \geq 0$ and $\chi = \alpha,\beta,\gamma,\delta$.

A.1 Transformation to integral equations

Define the characteristic lines as

$$\phi_i^\chi(x) = \int_0^x \frac{1}{\lambda_i^\chi(\xi)}d\xi, \quad \phi_i^\delta(x) = \int_x^1 \frac{1}{\lambda_i^\delta(\xi)}d\xi, \quad \text{(A.13)}$$

for $\chi = \alpha,\ \beta$,

$$\xi_i^\chi(x,t,s) = (\phi_i^\chi)^{-1}\left(\phi_i^\chi(x) + (s-t)\right) \quad \text{(A.14)}$$

for $\chi = \alpha,\ \beta,\ \delta$, and

$$s^0_{\chi,i}(x,t) = \max\{0, t - \phi_i^\chi(x)\} \quad \text{(A.15)}$$

for $\chi = \alpha,\ \beta,\ \delta$. Note that all ϕ_i^χ are strictly increasing and, hence, invertible. Therefore, for all (x,t), $\xi_i^\chi(x,t,s)$ is well-defined for all $s \in [s^0_{\chi,i}(x,t), t]$. Also note that for all $t \geq \phi_i^\chi(x)$,

$$\xi_i^\chi(x,t,s^0_{\chi,i}(x,t)) = 0, \qquad \chi = \alpha,\ \beta, \quad \text{(A.16)}$$
$$\xi_i^\chi(x,t,s^0_{\chi,i}(x,t)) = 1, \qquad \chi = \delta. \quad \text{(A.17)}$$

The PDEs (A.1)-(A.4) can be converted into integral equations by integrating them along the characteristic lines defined by (A.13)-(A.15). The integration paths for a point (x,t) in the special case $t=\phi_1^\delta(x)$ is sketched in Figure A.1. This gives

$$\alpha_i(x,t) = H_i^\alpha(x,t) + J_i^\alpha[\eta](x,t) + I_i^\alpha[\eta](x,t), \quad \text{(A.18)}$$
$$\beta_i(x,t) = H_i^\beta(x,t) + I_i^\beta[\eta](x,t), \quad \text{(A.19)}$$
$$\gamma_i(x,t) = H_i^\gamma(x,t) + I_i^\gamma[\eta](x,t), \quad \text{(A.20)}$$
$$\delta_i(x,t) = H_i^\delta(x,t) + I_i^\delta[\eta](x,t), \quad \text{(A.21)}$$

where the initial conditions (A.7) and state-independent boundary conditions in (A.5)-(A.6) are

$$H_i^\alpha(x,t) = \begin{cases} \alpha_{0,i}\left(\xi_i^\alpha(x,t,0)\right) & t < \phi_i^\alpha(x) \\ 0 & t \geq \phi_i^\alpha(x) \end{cases}, \quad \text{(A.22)}$$
$$H_i^\beta(x,t) = \begin{cases} \beta_{0,i}\left(\xi_i^\beta(x,t,0)\right) & t < \phi_i^\beta(x) \\ V_i^\beta\left(s_{\beta,i}^0(x,t)\right) & t \geq \phi_i^\beta(x) \end{cases}, \quad \text{(A.23)}$$
$$H_i^\gamma(x,t) = V_i^\gamma(t), \quad \text{(A.24)}$$
$$H_i^\delta(x,t) = \begin{cases} \delta_{0,i}\left(\xi_i^\delta(x,t,0)\right) & t < \phi_i^\delta(x) \\ V_i^\delta\left(s_{\beta,i}^0(x,t)\right) & t \geq \phi_i^\delta(x) \end{cases}, \quad \text{(A.25)}$$

the state-dependent boundary condition in (A.5) is

$$J_i^\alpha[\eta](x,t) = \begin{cases} 0 & t < \phi_i^\alpha(x) \\ g\left(\eta_0(s_{\alpha,i}^0(x,t)), s_{\alpha,i}^0(x,t)\right) & t \geq \phi_i^\alpha(x) \end{cases}, \quad \text{(A.26)}$$

and the coupling terms G^χ appear in

$$I_i^\chi[\eta](x,t) = \int_{s_{\chi,i}^0}^{t} G_i^\chi\left(\eta(\xi_i^\chi(x,t,s),s), \xi_i^\chi(x,t,s), s\right) ds,$$
$$\chi = \alpha, \beta, \delta, \quad \text{(A.27)}$$
$$I_i^\gamma[\eta](x,t) = \int_0^x G_i^\gamma\left(\eta(\xi,t), \xi, t\right) d\xi. \quad \text{(A.28)}$$

Finally, let H^χ, J^χ and I^χ be the vector-valued function with components as defined in (A.22)-(A.28).

A.2 Existence and boundedness of solutions

Theorem 25 *For bounded initial conditions and inputs and any $T>0$, and assuming (A.9)-(A.12), the integral equations (A.18)-(A.21) have a unique solution on $[0,1]\times[0,T]$ that satisfies the a-priory bound*

$$\sup_{x\in[0,1]} \|\eta(x,t)\| \leq c_1(t) c_2 e^{c_3 t} \quad \text{(A.29)}$$

for all $t \leq T$, where

$$c_1(t) = \sum_{\chi=\alpha,\beta,\delta} \|\chi_0\|_\infty + \sum_{\chi=\beta,\gamma,\delta} \sup_{s\in[0,t]} \|V^\chi(s)\|_\infty, \quad \text{(A.30)}$$

and the constants c_2 and c_3 depend on L_χ for $\chi=\alpha,\beta,\gamma,\delta,g$. Moreover, the solution depends Lipschitz-continuously on χ_0, $\chi=\alpha,\beta,\delta$ and V^χ, $\chi=\beta,\gamma,\delta$.

PROOF. The Lemma can be proven by use of a successive approximation argument similar to [23, Proposition 26]. Alternatively it can be proven by use of a contraction argument similar to [5, Theorem 3.1]. ■

A.3 Proof of Lemma 8

The following lemma is essential in the proof of Lemma 8

Lemma 26 *Consider the solution of (A.1)-(A.7). The states $\alpha(x,t)$ in the domain*

$$\mathcal{A}_\alpha = \left\{(x,t) : x\in[0,1],\, t \leq \phi_1^\delta(x)\right\} \setminus \{(0,\phi_1^\delta(0))\}, \quad \text{(A.31)}$$

$\beta(x,t)$ and $\gamma(x,t)$ in the domain

$$\mathcal{A}_{\beta\gamma} = \left\{(x,t) : x\in[0,1],\, t \leq \phi_1^\delta(x)\right\}, \quad \text{(A.32)}$$

$\delta_i(x,t)$ for $i=2,\ldots,n_\delta$ in the domain

$$\mathcal{A}_\delta = \left\{(x,t) : x\in[0,1),\, t \leq \phi_1^\delta(x)\right\}, \quad \text{(A.33)}$$

and $\delta_1(x,t)$ in the domain

$$\mathcal{A} = \left\{(x,t) : x\in[0,1],\, t < \phi_1^\delta(x)\right\}, \quad \text{(A.34)}$$

are independent of the input $V^\delta(t)$ for $t\geq 0$.

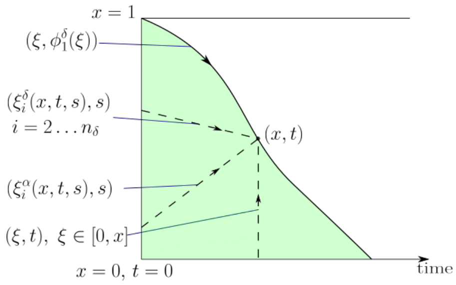

Fig. A.1. Integration paths of integral equations (A.18)-(A.21) at a point (x,t) with $t=\phi_1^\delta(x)$. The area shaded in green indicates the domain $\mathcal{A}$ (see also Equations (A.31)-(A.34)).

PROOF. The integration paths for the integral equations for a point (x,t) with $t=\phi_1^\delta(x)$ is shown in Figure A.1. Briefly speaking, on the respective sets all terms in the integral equations (A.18)-(A.21) are independent of the input V^δ. See also [23, Appendix A] for a detailed proof for the case $m=n=1$, and [5, Chapter 3] for a more general discussion of determinate sets. ■

Note that at the solution at points $\alpha(0,\phi_1^\delta(0))$ depends on δ_1, i.e. on V^δ, via the boundary condition (A.5), while the boundary value $\delta(1,0)$ is V^δ by definition. Therefore, these points are exempted from the sets $\mathcal{A}_\alpha$ and $\mathcal{A}_\delta$, respectively.

A.4 Proof of Theorems 12 and 22

The following lemma about convergence to the origin is used in Theorems 12 and 22.

Lemma 27 *Consider (A.1)-(A.7) for $n_\delta = 0$ and assume $\lambda^\alpha_{n_\alpha}(x) < \lambda^\beta_{n_\beta}(x)$ for all. If $V^\beta(t) = V^\gamma(t) = 0$ for all $t \geq 0$, then $\eta(x,t) = 0$ for all $x \in [0,1]$, $t \geq \phi^\alpha_{n_\alpha}(x)$.*

PROOF. One can show that under the given assumptions, $t \geq \phi^\alpha_{n_\alpha}(x)$ implies

$$
\begin{aligned}
s^0_{\chi,i}(x,t) \geq 0 \quad & \forall \chi = \alpha, \beta,\, i = 1, \ldots, n_\chi, \quad \text{(A.35)}\\
\phi^\alpha_{n_\alpha}(\xi^\chi_i(x,t,s)) \leq s \quad & \forall s \in [s^0_{\chi,i}(x,t), t],\, \chi = \alpha, \beta,\\
& \quad i = 1, \ldots, n_\chi, \quad \text{(A.36)}\\
\phi^\alpha_{n_\alpha}(\xi) \leq t \quad & \forall \xi \in [0,x]. \quad \text{(A.37)}
\end{aligned}
$$

Because of (A.36)-(A.37), the domain $\mathcal{B} = \{(x,t) : x \in [0,1],\, t \geq \phi^\alpha_{n_\alpha}(x)\}$ is a determinate set, i.e. for $(x,t) \in \mathcal{B}$ all state-dependent terms in the integral equations (A.18)-(A.21) depend on the state evaluated at points in $\mathcal{B}$ only. Because of (A.35) and (A.5), $H^\alpha(x,t) = H^\beta(x,t) = H^\gamma(x,t) = 0$ for $t \geq \phi^\alpha_{n_\alpha}(x)$. Moreover, there is no δ and H^δ. Therefore, one can see that for points $t \geq \phi^\alpha_{n_\alpha}(x)$ the zero-solution solves the integral equations (A.18)-(A.21). Uniqueness of the solution completes the proof. ■

A.5 Proof of Lemma 20

The following lemma, which is tailored to prove Lemma 20, can be proven using similar ideas as in Lemma 26. Define

$$\hat{\phi}^\alpha_i(x) = -\int_x^1 \frac{1}{\lambda^\alpha_i(\xi)} d\xi. \quad \text{(A.38)}$$

Lemma 28 *For $n_\beta = n_\gamma = 0$, $j = 1, \ldots, n$ and given $(\hat{\alpha}_0, \hat{\delta}_0)$, system (A.1)-(A.6) with the modified initial conditions*

$$\alpha(x, \hat{\phi}_{\alpha,j}(x) = \hat{\alpha}_0(x), \quad \delta(x, \hat{\phi}_{\alpha,j}(x) = \hat{\delta}_0(x), \quad \text{(A.39)}$$

has a unique solution in the determinate set

$$\mathcal{C} = \left\{(x,t) : x \in [0,1),\, t \in [\hat{\phi}_{\alpha,j}(x), \phi^\delta_1(x))\right\} \quad \text{(A.40)}$$

that is independent of $V^\delta(t)$ for $t \geq 0$.